MPBP with continuous intermediate variables

Applied to Glauber dynamics

Federico Florio

December 2024

1 Convolution of tensor trains

Take N probability distributions, each one of a vector of length T of spins $x_i^t \in \{-1,+1\}$, written in tensor train form (can be approximations): $p_i(\overline{x}_i) = A_i^1(x_i^1)\,A_i^2(x_i^2)\dots A_i^T(x_i^T)\ \forall i \in 1\dots N$.
It is wanted to calculate the probability distribution of the variable $h \coloneqq \sum_{k=1}^N J_k \overline{x}_k$, that is $h^t = \sum_{k=1}^N J_k \overline{x}_k^t\ \forall t \in 1\dots T$.[1]

$$
\begin{aligned}
p(y^1,\dots,y^T) &= \sum_{\overline{\boldsymbol{x}}} \prod_i p_i(\overline{x}_i) \prod_t \delta_{y^t,\sum_k J_k x_k^t} = \\
&= \sum_{\overline{\boldsymbol{x}}} \prod_i \prod_t A_i^t(x_i^t) \prod_t \delta_{y^t,\sum_k J_k x_k^t} = \\
&= \sum_{\overline{\boldsymbol{x}}} \prod_i \sum_{\overline{m}_i} \prod_t A_i^t(x_i^t)_{m_i^t,m_i^{t+1}} \prod_t \delta_{y^t,\sum_k J_k x_k^t} = \\
&= \sum_{\overline{\boldsymbol{x}}} \sum_{\overline{\boldsymbol{m}}} \prod_t \left(\delta_{y^t,\sum_k J_k x_k^t} \prod_i A_i^t(x_i^t)_{m_i^t,m_i^{t+1}} \right) = \\
&= \sum_{\overline{\boldsymbol{m}}} \prod_t \sum_{\boldsymbol{x}^t} \left(\delta_{y^t,\sum_k J_k x_k^t} \prod_i A_i^t(x_i^t)_{m_i^t,m_i^{t+1}} \right)
\end{aligned}
$$

Now, the tensors can be expanded in a basis, going from a discrete space to a continuous one. In practice, the elements $A_i^t(x_i^t)_{m_i^t,m_i^{t+1}}$ are written as $\sum_\alpha F_i^t\,[\alpha]_{m_i^t,m_i^{t+1}} u_\alpha(x_i^t)$. In the present case, the basis functions u_α are chosen to be plane waves with some period P (so that the expansion is a truncated Fourier series): $u_\alpha(x) = e^{i\frac{2\pi}{P}\alpha x}$ (see section 2 for more details). Then:

$$
p(y^1,\dots,y^T) = \sum_{\overline{\boldsymbol{m}}} \prod_t \sum_{\boldsymbol{x}^t} \left(\delta_{y^t,\sum_k J_k x_k^t} \prod_i \sum_\alpha F_i^t\,[\alpha]_{m_i^t,m_i^{t+1}} u_\alpha(x_i^t) \right)
$$

and promoting the x_i^ts to be continuous variables:

$$
p(y^1,\dots,y^T) = \sum_{\overline{\boldsymbol{m}}} \prod_t \int \mathrm{d}x_1^t\,\mathrm{d}x_2^t \dots \mathrm{d}x_N^t \left(\delta\left(y^t - \sum_k J_k x_k^t \right) \prod_i \sum_\alpha F_i^t[\alpha]_{m_i^t,m_i^{t+1}} u_\alpha(x_i^t) \right)
$$

In order to close the matrix-product ansatz and recover a tensor-train form (expanded in Fourier basis) for $p(y^1,\dots,y^T)$, the following form is sought:

$$
\begin{aligned}
p(y^1,\dots,y^T) &= \prod_t B^y(y^t) = \sum_{\overline{n}} \prod_t B^t(y^t)_{n^t,n^{t+1}} \\
&= \sum_{\overline{n}} \prod_t \sum_\gamma G^t[\gamma]_{n^t,n^{t+1}} u_\gamma(y^t)
\end{aligned}
$$

[1] actually, if $x_k^t \in [-P/2,+P/2]$, in order to leave the domain unaltered, the rescaling $x_k^t \mapsto x_k^t/N$ is applied, so that $x_k^t \in [-P/2N,+P/2N]$ and $y^t \in [-P/2,+P/2]$.

It can be easily noticed that:

$$B^t(y^t)_{n^t,n^{t+1}} = \int \mathrm{d}x_1^t\,\mathrm{d}x_2^t\ldots\mathrm{d}x_N^t \left(\delta\left(y^t, \sum_k J_k x_k^t\right)\prod_i\sum_\alpha F_i^t[\alpha]_{m_i^t,m_i^{t+1}}u_\alpha(x_i^t)\right)$$

where the index n^t is actually a multi-index $n^t = (m_1^t, m_2^t, \ldots, m_N^t)$.
Then, in the case of $N = 2$ tensors (which is the recursive step that allows to calculate all the others):

$$\begin{aligned}
B^t(y^t)_{(m_1^t,m_2^t),(m_1^{t+1},m_2^{t+1})} &= \int \mathrm{d}x_1^t \int \mathrm{d}x_2^t \left[\delta(y^t - J_1x_1^t - J_2x_2^t)\left(\sum_\alpha F_1^t[\alpha]_{m_1^t,m_1^{t+1}}\,u_\alpha(x_1^t)\right)\left(\sum_\beta F_2^t[\beta]_{m_2^t,m_2^{t+1}}\,u_\beta(x_2^t)\right)\right] = \\
&= \sum_{\alpha,\beta} F_1^t[\alpha]_{m_1^t,m_1^{t+1}}F_2^t[\beta]_{m_2^t,m_2^{t+1}}\int \mathrm{d}x_1^t \int \mathrm{d}x_2^t\delta(y^t - J_1x_1^t - J_2x_2^t)u_\alpha(x_1^t)u_\beta(x_2^t)
\end{aligned}$$

And then it is easy to find the coefficients of the basis expansion:

$$\begin{aligned}
G^t[\gamma]_{(m_1^t,m_2^t),(m_1^{t+1},m_2^{t+1})} &= \frac{1}{P}\int B^t(y^t)_{(m_1^t,m_2^t),(m_1^{t+1},m_2^{t+1})}u_\gamma^*(y^t)\,\mathrm{d}y^t = \\
&= \frac{1}{P}\sum_{\alpha,\beta} F_1^t[\alpha]_{m_1^t,m_1^{t+1}}F_2^t[\beta]_{m_2^t,m_2^{t+1}}\int \mathrm{d}x_1^t\int \mathrm{d}x_2^t\int \mathrm{d}y^t\,\left[\delta(y^t - J_1x_1^t - J_2x_2^t)u_\alpha(x_1^t)u_\beta(x_2^t)u_\gamma^*(y^t)\right] = \\
&= \frac{1}{P}\sum_{\alpha,\beta} F_1^t[\alpha]_{m_1^t,m_1^{t+1}}F_2^t[\beta]_{m_2^t,m_2^{t+1}}\int \mathrm{d}x_1^t\int \mathrm{d}x_2^t\,\left[u_\alpha(x_1^t)u_\beta(x_2^t)u_\gamma^*(J_1x_1^t + J_2x_2^t)\right]
\end{aligned}$$

All that remains is to calculate the double integral:

$$\begin{aligned}
I_{\alpha,\beta,\gamma} &= \frac{1}{P}\int \mathrm{d}x_1^t\int \mathrm{d}x_2^t\,\left[u_\alpha(x_1^t)u_\beta(x_2^t)u_\gamma^*(J_1x_1^t + J_2x_2^t)\right] = \\
&= \frac{1}{P}\int \mathrm{d}x_1 e^{i\frac{2\pi}{P}(\alpha - J_1\gamma)x_1}\int \mathrm{d}x_2 e^{i\frac{2\pi}{P}(\beta - J_2\gamma)x_2} = \\
&= \frac{1}{P}\left(-\frac{iP}{2\pi(\alpha - J_1\gamma)}\left[e^{i\frac{2\pi}{P}(\alpha - J_1\gamma)x_1}\right]_{x_1=-s_1}^{x_1=+s_1}\right)\cdot\left(-\frac{iP}{2\pi(\beta - J_2\gamma)}\left[e^{i\frac{2\pi}{P}(\beta - J_2\gamma)x_2}\right]_{x_2=-s_2}^{x_2=+s_2}\right) = \\
&= \frac{4}{P}\frac{\sin\left(\frac{2\pi}{P}(\alpha - J_1\gamma)s_1\right)}{\frac{2\pi}{P}(\alpha - J_1\gamma)}\cdot\frac{\sin\left(\frac{2\pi}{P}(\beta - J_2\gamma)s_2\right)}{\frac{2\pi}{P}(\beta - J_2\gamma)}
\end{aligned}$$

Then the desired form is recovered:

$$G^t[\gamma]_{(m_1^t,m_2^t),(m_1^{t+1},m_2^{t+1})} = \sum_{\alpha,\beta} F_1^t[\alpha]_{m_1^t,m_1^{t+1}}F_2^t[\beta]_{m_2^t,m_2^{t+1}}I_{\alpha,\beta,\gamma}$$

and one can calculate the convolution with all the N terms in a recursive way, by taking the right s_1 and s_2. The only problem is that, at each iteration, the sizes of the tensors grow, rendering this approach of exponential complexity in N. This can be solved by applying, at each iteration, two sweeps of SVD and truncation to the tensors $G^t[\gamma]$, reducing and controlling their sizes.

2 Calculation of coefficients

One has a probability distribution for a spin: $p(x) = p(-s_1)\delta_{x,-s_1} + p(+s_1)\delta_{x,+s_1}$ and wants to compute its Fourier series coefficients. Thus, the distribution can be approximated with a sum of Gaussian, each with small variance σ^2, as depicted in Figure 1:

$$f(x) \simeq \frac{1}{\sqrt{2\pi\sigma^2}}\left(p(-s_1)e^{-\frac{(x+s_1)^2}{2\sigma^2}} + p(+s_1)e^{-\frac{(x-s_1)^2}{2\sigma^2}}\right)$$

The coefficients of the Fourier series can then be calculated as scalar products with the basis functions:

$$\begin{aligned}
C_\alpha &= \frac{1}{P}\int_{-P/2}^{+P/2} f(x)u_\alpha^*(x)\,\mathrm{d}x = \frac{1}{P}\int_{-P/2}^{+P/2}\frac{1}{\sqrt{2\pi\sigma^2}}\left(p(-s_1)e^{-\frac{(x+s_1)^2}{2\sigma^2}} + p(+s_1)e^{-\frac{(x-s_1)^2}{2\sigma^2}}\right)e^{i\frac{2\pi}{P}\alpha x}\,\mathrm{d}x \simeq \\
&\simeq \frac{1}{P\sqrt{2\pi\sigma^2}}\left(p(-s_1)\int_{-\infty}^{+\infty} e^{-\frac{(x+s_1)^2}{2\sigma^2} - i\frac{2\pi}{P}\alpha x}\,\mathrm{d}x + p(+s_1)\int_{-\infty}^{+\infty} e^{-\frac{(x-s_1)^2}{2\sigma^2} - i\frac{2\pi}{P}\alpha x}\,\mathrm{d}x\right)
\end{aligned}$$

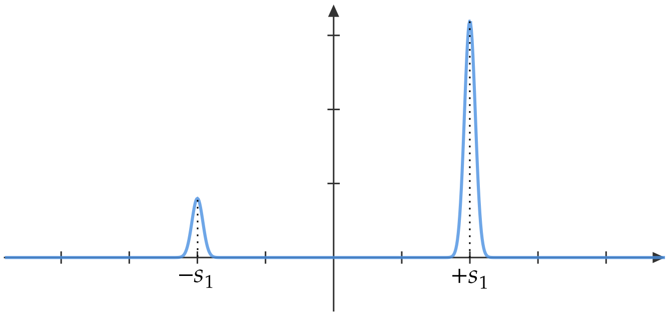

Figure 1: Approximation of the pdf of a spin as a sum of two Gaussians.

The integrals are easily calculated by completing the squares in the exponents:

$$\int_{-\infty}^{+\infty} e^{-\frac{(x\pm s_1)^2}{2\sigma^2}-i\frac{2\pi}{P}\alpha x}\,\mathrm{d}x = \sqrt{2\pi\sigma^2}\;e^{-\frac{2\pi^2}{P^2}\sigma^2\alpha^2}\left[\cos\left(\frac{2\pi}{P}\alpha s_1\right)\pm i\sin\left(\frac{2\pi}{P}\alpha s_1\right)\right]$$

$$\begin{aligned}
C_\alpha &= \frac{e^{-\frac{2\pi^2}{P^2}\sigma^2\alpha^2}}{P}\left[p(-s_1)\left(\cos\left(\frac{2\pi}{P}\alpha s_1\right)+i\sin\left(\frac{2\pi}{P}\alpha s_1\right)\right)+p(+s_1)\left(\cos\left(\frac{2\pi}{P}\alpha s_1\right)-i\sin\left(\frac{2\pi}{P}\alpha s_1\right)\right)\right]= \\
&= \frac{e^{-\frac{2\pi^2}{P^2}\sigma^2\alpha^2}}{P}\left[(p(-s_1)+p(s_1))\cos\left(\frac{2\pi}{P}\alpha s_1\right)+i(p(-s_1)-p(s_1))\sin\left(\frac{2\pi}{P}\alpha s_1\right)\right]= \\
&= \frac{e^{-k_\alpha^2\sigma^2}}{P}\left[(p(-s_1)+p(s_1))\cos\left(k_\alpha s_1\right)+i(p(-s_1)-p(s_1))\sin\left(k_\alpha s_1\right)\right]
\end{aligned}$$

having defined $k_\alpha \coloneqq \frac{2\pi}{P}\alpha$.

3 Belief Propagation message update

The BP update equations are to be written using the tensor-train approximation and the decomposition on a Fourier basis.

$$\begin{aligned}
\mu_{i\to j}(\overline{x}_i,\overline{x}_j) &= \sum_{\{\overline{x}_k\}_{k\in\partial i\setminus j}}\prod_t \omega\left(x_i^{t+1},\sum_{k\in\partial i}J_{ki}x_k^t\right)\prod_{k\in\partial i\setminus j}\mu_{k\to i}(\overline{x}_k,\overline{x}_i)= \\
&= \sum_{\{\overline{x}_k\}_{k\in\partial i\setminus j}}\int \mathrm{d}y^0\cdots\int\mathrm{d}y^T\prod_t\omega\left(x_i^{t+1},y^t+J_{ji}x_j^t\right)\delta\left(y^t-\sum_{k\in\partial i\setminus j}J_{ki}x_k^t\right)\prod_{k\in\partial i\setminus j}\mu_{k\to i}(\overline{x}_k,\overline{x}_i)= \\
&= \int\cdots\int\prod_t\mathrm{d}y^t\prod_t\omega\left(x_i^{t+1},y^t+J_{ji}x_j^t\right)\underbrace{\sum_{\{\overline{x}_k\}_{k\in\partial i\setminus j}}\prod_t\delta\left(y^t-\sum_{k\in\partial i\setminus j}J_{ki}x_k^t\right)\prod_{k\in\partial i\setminus j}\mu_{k\to i}(\overline{x}_k,\overline{x}_i)}_{\text{this is a convolution of the type described in section 1}}= \\
&= \int\cdots\int\prod_t\mathrm{d}y^t\prod_t\omega\left(x_i^{t+1},y^t+J_{ji}x_j^t\right)\prod_t B^t(y^t,x_i^t)= \\
&= \prod_t\int\mathrm{d}y^t\omega\left(x_i^{t+1},y^t+J_{ji}x_j^t\right)\sum_{\gamma^t}G^t[\gamma^t,x_i^t]u_{\gamma^t}(y^t)
\end{aligned}$$

Then, for each time:

$$\tilde{A}^t(x_i^{t+1},x_i^t,x_j^t)=\int\mathrm{d}y^t\omega\left(x_i^{t+1},y^t+J_{ji}x_j^t\right)\sum_{\gamma^t}G^t[\gamma^t,x_i^t]u_{\gamma^t}(y^t)$$

where the calculation has to be performed for each and every matrix element:

$$\begin{aligned}\left[\tilde{A}^t(x_i^{t+1},x_i^t,x_j^t)\right]_{n^t,n^{t+1}} &= \int \mathrm{d}y^t \omega\left(x_i^{t+1}, y^t + J_{ji}x_j^t\right) \sum_{\gamma^t} \left[G^t[\gamma^t, x_i^t]\right]_{n^t,n^{t+1}} u_{\gamma^t}(y^t) = \\ &= \sum_{\gamma^t} \left[G^t[\gamma^t, x_i^t]\right]_{n^t,n^{t+1}} \int \mathrm{d}y^t \omega\left(x_i^{t+1}, y^t + J_{ji}x_j^t\right) u_{\gamma^t}(y^t) = \\ &= \sum_{\gamma^t} \left[G^t[\gamma^t, x_i^t]\right]_{n^t,n^{t+1}} I_{\gamma^t}(x_i^{t+1}, x_j^t)\end{aligned}$$

The indefinite integral can be calculated as follows[2]:

$$\begin{aligned}I_{\gamma^t}(x_i^{t+1},x_j^t) &= \int \mathrm{d}y^t \omega\left(x_i^{t+1}, y^t + J_{ji}x_j^t\right) u_{\gamma^t}(y^t) = \\ &= \int \mathrm{d}y^t \frac{e^{\beta(y^t+J_{ji}x_j^t+h_i)x_i^{t+1}}}{2\cosh(\beta(y^t+J_{ji}x_j^t+h_i))} e^{i\frac{2\pi}{P}\gamma^t y^t} = \\ &= \frac{e^{ik_\gamma y^t}}{\beta(1+x_i^{t+1})+ik_\gamma} e^{\beta(y^t+J_{ji}x_j^t+h_i)(1+x_i^{t+1})} {}_2F_1\left(1, \frac{\beta(1+x_i^{t+1})+ik_\gamma}{2\beta}, \frac{\beta(3+x_i^{t+1})+ik_\gamma}{2\beta}, -e^{2\beta(y^t+J_{ji}x_j^t+h_i)}\right)\end{aligned}$$

and the extrema for calculating the definite integral are $y^t = -d_i + 1$ and $y^t = d_i - 1$, respectively.
If $x_i^{t+1} = -1$ and $\gamma = 0$ the integral can be easily calculated to be:

$$I_0(-1, x_j^t) = \int \mathrm{d}y^t \frac{e^{-\beta(y^t+J_{ji}x_j^t+h_i)}}{e^{\beta(y^t+J_{ji}x_j^t+h_i)} + e^{-\beta(y^t+J_{ji}x_j^t+h_i)}} = -\frac{1}{2\beta}\log\left(1+e^{-2\beta(y^t+J_{ji}x_j^t+h_i)}\right)$$

The final sought form is $\mu_{i\to j}(\overline{x}_i, \overline{x}_j) = \prod_t A(x_i^t, x_j^t)$. In the previous form, the variables x_i^{t+1} and x_i^t were coupled in the same tensor $\tilde{A}^t$. To solve this, a sweep of SVD is sufficient.

4 Calculation of beliefs

The calculations are essentially the same as those performed in section 3 for the messages. The only difference is that the convolution is performed on all $\mu_{k\to i} \forall k \in \partial i$ and that the summation is performed on $\{\overline{x}_k\}_{k\in\partial i}$. These facts have the effect that the integral $I_{\gamma^t}(x_i^{t+1}, x_j^t)$ does not depend on x_j^t, so one can formally define a new integral $I'_{\gamma^t}(x_i^{t+1}) := I_{\gamma^t}(x_i^{t+1}, x_j^t = 0)$. Then the beliefs can be computed:

$$b_i(\overline{x}_i) = \prod_t C^t(x_i^t)$$

where the matrices C^t are obtained by means of a sweep of SVD from:

$$\tilde{C}^t(x_i^t) = \sum_{\gamma^t} G^t[\gamma^t, x_i^t] I'_{\gamma^t}(x_i^{t+1})$$

with G^t being the matrices resulting from the convolution of all the incoming messages in i.

[2]This has been done by Wolfram Mathematica with the input `Integrate[1/2 Exp[b (y + J) x]/Cosh[b (y + J)] Exp[I k y], y, Assumptions -> {b \in Reals, J \in Reals, x \in Reals, k \in Reals, y \in Reals}]`.